

PROGRAMMATORE DI EPROM

KIT <i>Service</i>	
Difficoltà	⚡ ⚡
Tempo	⌚
Costo	L.26.000

Pochi componenti standard bastano per poter programmare la maggior parte dei tipi di EPROM con il vostro PC od un qualsiasi altro computer. Questo utilissimo circuito può essere vantaggiosamente combinato all'altrettanto semplice lettore di EPROM che presenteremo il prossimo mese.

Se vi capita soltanto occasionalmente di dover programmare o duplicare EPROM, questo circuito fa per voi. Non è stato invece assolutamente progettato per una produzione di serie: se tentaste di utilizzarlo in questo senso, ne rimarreste delusi.

Uno strumento sempre più necessario

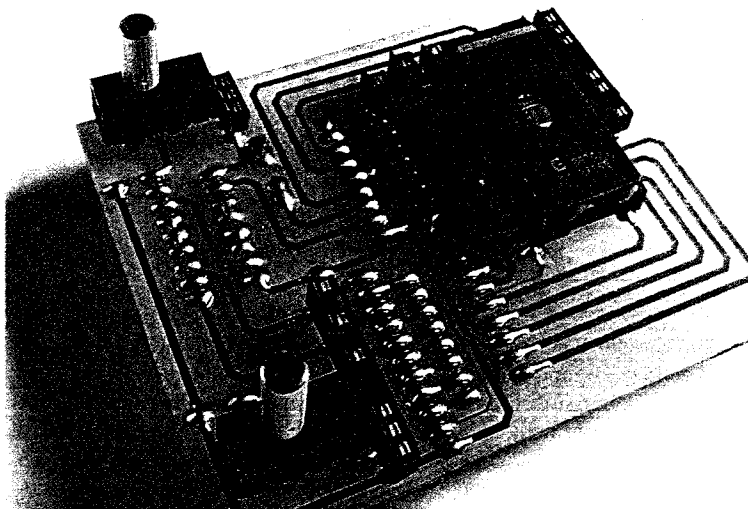
Chiunque si occupi di elettronica, anche a livello amatoriale, può trovarsi facilmente nella necessità di utilizzare le EPROM (memorie di sola lettura, cancellabili mediante raggi ultravioletti). Contrariamente ad una convinzione molto diffusa, questi componenti non servono soltanto ad immagazzinare programmi per microprocessori, ed inoltre da qualche anno il loro prezzo continua a diminuire. Rinunciare ad utilizzare le EPROM per la mancanza di un opportuno programmatore vuol dire privarsi di possibilità davvero allettanti.

D'altra parte, è chiaro che un dilettante ben difficilmente investe una cifra abbastanza rilevante nell'acquisto di uno strumento che gli servirà forse quattro o

cinque volte all'anno. I programmatori di EPROM professionali sono molto costosi, perché devono risultare semplici da utilizzare, nonostante l'enorme varietà dei componenti programmabili (un produttore ne ha elencati 1800 tipi). Inoltre questi strumenti devono essere anche molto rapidi, perché vengono impiegati quasi sempre nelle linee di produzione. Il riconoscimento automatico

modifica del contenuto della EPROM, oppure la programmazione a partire da un listato, richiedono una lunga e fastidiosa manovra di interruttori: immaginiamo ad esempio che una 2764 contenga 8192 "indirizzi", ciascuno dei quali occupa 8 bit.

La nostra geniale soluzione consiste nell'equipaggiare un microcomputer con un accessorio "programmatore di E-



dei vari tipi di componenti, gli algoritmi di adattamento ed altre sofisticazioni rendono questi dispositivi completamente inaccessibili al singolo, a meno che egli faccia programmare le sue memorie da un professionista (ma a che prezzo!). I produttori di EPROM inoltre, esprimono serie riserve sui programmatori, anche "professionali", che non hanno ricevuto la loro approvazione.

Il dilettante deve generalmente accontentarsi di strumenti alquanto rozzi, in grado di programmare una EPROM in modo abbastanza soddisfacente ricopiando una EPROM dello stesso tipo che serve da "modello". D'altro canto, la

PROM": un programma abbastanza efficiente può mettere al servizio dell'utilizzatore esigente la straordinaria potenza di elaborazione dei computer attuali, così da soddisfare tutte le sue segrete ambizioni.

Questo tipo di accessorio, anche se molto meno costoso di un programmatore completo, è spesso ancora complicato e dispendioso, forse perché gli si richiede un po' troppo.

Una soluzione economica

Fondamentalmente, programmare una EPROM è un'operazione semplice: basta selezionare ciclicamente gli indirizzi

zi nei quali si desidera scrivere e poi, in presenza del dato da programmare (normalmente una parola da 8 bit), inviare un determinato impulso ad un particolare piedino. Naturalmente, il tutto deve avvenire in presenza delle tensioni di alimentazione specificate dal produttore. I dati da programmare possono essere

tamento della programmazione (a meno che, ovviamente, non debba produrre 200 pezzi!)

La nostra idea consiste dunque nell'utilizzare, per il collegamento del programmatore, il connettore per stampante parallela ("CENTRONICS") di cui sono forniti praticamente tutti i compu-

mo; più raramente in modo parziale, ma sempre sequenzialmente, nell'ordine crescente degli indirizzi. Un semplice contatore, che utilizzi gli impulsi di programmazione come segnale di clock, potrà facilmente "esplorare" tutti gli stati possibili del bus degli indirizzi, purché venga azzerato in sincronismo con il programma, cosa per niente difficoltosa.

Per quanto riguarda le tensioni di alimentazione, ogni laboratorio normalmente fornito contiene quanto è necessario per produrre una tensione di 5 V ed una seconda tensione, variabile tra 12 e 25 V, con assorbimento di qualche decina di milliampere. Al limite, si potrà costruire un piccolo alimentatore "dedicato", ma non è indispensabile.

Passiamo alla pratica

La Figura 1 illustra nei particolari la soluzione pratica scelta: un contatore CMOS 4040 ed un invertitore sestuplo 4049 sono i soli due componenti attivi necessari!

Otto resistori da 470 Ω (oppure una rete DIL) sono inseriti sul bus dei dati, per evitare ogni danno in caso di "conflitto": essi limitano la corrente che potrebbe circolare se la memoria venisse acciden-

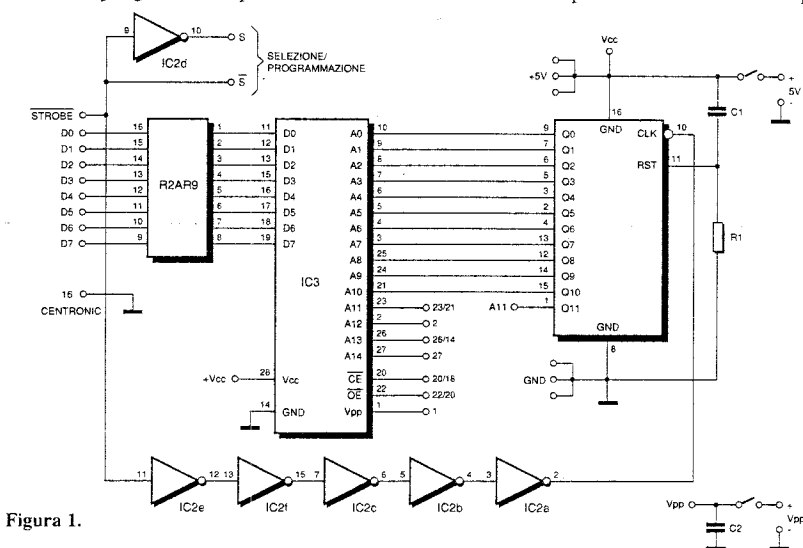


Figura 1.

disponibili in un'altra EPROM (non necessariamente dello stesso tipo di quella che viene programmata), su dischetto, su cassetta, su carta oppure nella testa dell'operatore.

Complica le cose il fatto che, da un tipo di EPROM all'altro, variano la piedinatura, la tensione d'alimentazione ed il ciclo di programmazione, con differenze spesso non irrilevanti.

Se richiediamo che il nostro programmatore sia "intelligente", e quindi tenga conto di tutti i casi possibili, dovrà essere un dispositivo complesso e configurabile via software.

Volendo che sia anche veloce, non c'è altra soluzione che interfacciarlo direttamente al bus degli indirizzi e dei dati del computer ospite, mediante circuiti del tipo "PIA".

A nostro avviso, un elettronico degno di tale nome è senz'altro in grado di collegare qualche filo senza sbagliarsi e può attendere qualche minuto per il comple-

ter: le otto linee dei dati permettono di trasmettere direttamente i byte da programmare, mentre la linea di STROBE può facilmente convogliare gli impulsi di programmazione, la cui durata è regolabile via software (non è cioè necessario un monostabile di precisione).

Sono a disposizione inoltre anche alcune altre linee d'ingresso (almeno una linea ACK od una BUSY): saranno utili in un secondo tempo, per leggere le nostre EPROM, per un'eventuale duplicazione, verifica o, perché no, modifica.

Non è tuttavia possibile far passare per questa via troppo "stretta" anche un bus degli indirizzi, ma la cosa non è importante; nella stragrande maggioranza dei casi, le EPROM si programmano a partire dal loro primo indirizzo fino all'ulti-

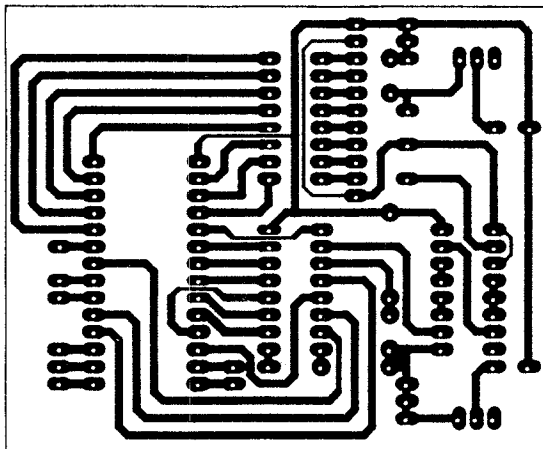
PROGRAMMAZIONE	PROGRAMMAZIONE		
	A14	A13	A12
BLOCCO 1		GND	
BLOCCO 2	GND	GND	+5V
BLOCCO 3	GND	+5V	GND
BLOCCO 4	GND	+5V	+5V
BLOCCO 5	+5V	GND	GND
BLOCCO 6	+5V	GND	+5V
BLOCCO 7	+5V	+5V	GND
BLOCCO 8	+5V	+5V	+5V

Figura 2.

talmente disposta in modo "lettura". Le otto linee dei dati, così protette, vengono collegate direttamente allo zoccolo della EPROM, perché la loro collocazione è la stessa per tutte le memorie conformi allo standard "BYTEWIDE".

Il segnale di STROBE ed il suo complemento sono disponibili per l'utilizzatore su due piedini, in modo che un impulso della polarità corretta possa essere inviato alla memoria, a seconda del suo tipo (positivo per le 2716, negativo per la maggior parte degli altri tipi). Questo stesso segnale di STROBE, complementato e ritardato da cinque invertitori collegati in cascata, viene applicato all'ingresso di clock del 4040, il cui azzeramento è automatico all'applicazione della tensione (rete R1-C1). Scopo di questo ritardo è di impedire all'indirizzo di cambiare prima che scompaia l'impulso di programmazione. Dodici linee di indirizzamento sono di-

Figura 3.



sponibili all'uscita del 4040, e sono sufficienti per memorie che arrivano fino a 4 Kbyte ($2^{12} = 4096$), cioè fino alla 2732. Le linee da A0 a A10, comuni a tutte le EPROM dalla 2716 alla 27256 ed oltre, sono collegate direttamente allo zoccolo. A1 viene fornita su un terminale, onde permettere il suo collegamento "volante" ai piedini 23 o 21 dello zoccolo, a partire dalla EPROM 2732. Riteniamo che la 2716 e le 2732 siano ampiamente sufficienti per i dilettanti (non è raro che le 2764 restino vuote più della metà), ma abbiamo voluto che il nostro strumento potesse fare, in caso di

necessità, qualcosa di più. Per utilizzare le linee A12-A14 occorre utilizzare i piedini 1, 2, 27 e 28 dello zoccolo, che restavano inutilizzati con le 2716 e 2732. In Figura 2 si vede che, con spostamenti relativamente semplici dei conduttori di collegamento, si possono programmare le 2764 in due blocchi di 4 Kbyte; con un po' più di pazienza si possono anche programmare le 27128 in quattro blocchi e le 27256 in otto blocchi. Questo procedimento sarebbe ovviamente troppo mac-

chinoso per un utilizzo regolare, ma può essere utile prevedere questa opportunità, nel caso si presentasse l'occasione, acquistando uno zoccolo a 28 piedini invece di uno a 24. Due interruttori permettono di applicare e di interrompere le due tensioni di alimentazione Vcc e Vpp, secondo le indicazioni del software: ricordiamo infatti che la presenza di Vpp, in assenza di Vcc, risulta fatale per la memoria in una frazione di secondo. Siamo certi, comunque, che sarete in grado di vigilare affinché la procedura venga rispettata.

Collegamento della scheda

Un circuito così semplice trova posto senza difficoltà su una basetta incisa secondo il tracciato di Figura 3. L'originalità sta nel fatto che il lato componenti (Figura 4) è sfruttato abbastanza poco:

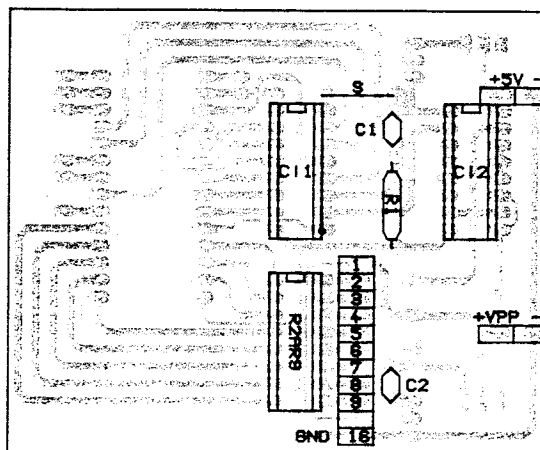
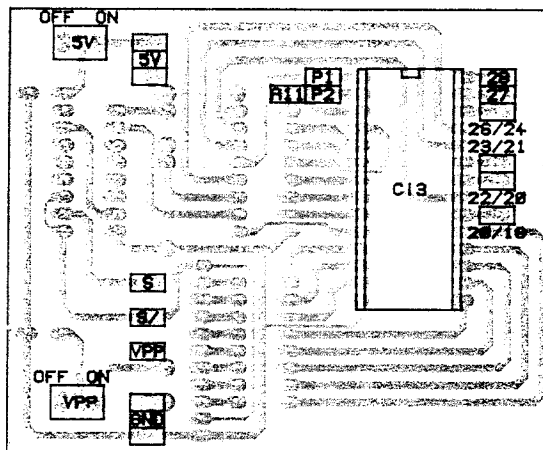


Figura 4.

Figura 5.



vi si trovano i due circuiti integrati, i resistori, i condensatori, un ponticello e la partenza dei conduttori (alimentazioni Vcc e Vpp e piattina a dieci conduttori proveniente dalla spina normalizzata CENTRONICS). E' sul lato rame (Figura 5) che verranno saldati lo zoccolo a 28 piedini (di buona qualità e possibilmente a forza di inser-

zione zero) e soprattutto i 18 spinotti a saldare, in barretta oppure recuperati da un vecchio zoccolo per circuito integrato.

Questi spinotti serviranno a determinare la configurazione da dare allo zoccolo: invece di conduttori muniti di spinotti a banana, utilizzare spezzoni di filo rigido per cablaggio da 6/10 (il diametro deve essere proprio questo) spellati a ciascuna estremità per 3 o 4 mm. Questa soluzione è molto meno ingombrante, ma altrettanto pratica; per non parlare del prezzo, che è imbattibile.

Il vantaggio di tale disposizione è che la scheda può essere fissata sul pannello superiore del contenitore, utilizzando semplici dadi come distanziali: lo zoccolo, gli spinotti e gli interruttori affiorano in modo piacevole, pratico e sicuro.

Per il momento, non continuiamo a parlare dell'inserimento nel contenitore, perché il prossimo mese descriveremo un lettore di EPROM di analogia concezione: è logico riunire i due apparecchi in uno stesso contenitore e collegarli al medesimo connettore.

Software

Senza un computer "ospite" ed un adatto programma, questo circuito è del tutto inutilizzabile.

Il nostro prototipo funziona con un computer PC compatibile (Commodore PC1), per il quale è stato scritto il programma di Figura 6. In ogni caso, visto che la piedinatura dei connettori CENTRONICS è normalizzata, quasi tutti gli home computer andranno altrettanto bene, ad eccezione degli AMSTRAD CPC, sui quali il costruttore ha pensato bene di non collegare l'ottava linea dei dati (l'inconveniente comunque si può risolvere).

Descrivendo in dettaglio il funzionamento della routine principale del programma, la Figura 7 permette di procedere alle modifiche necessarie per adat-

```

10 REM ---- PROGRAM ----
20 CLS
30 PRINT"INTERROMPERE LE ALIMENTAZIONI
40 PRINT"nome del file da riversare su EPROM?
50 INPUT F$:F$=F$+" ROM"
60 OPEN"1",#1,F$
70 DIM M(4096)
80 PRINT:PRINT"---- LETTURA FILE IN CORSO ----
90 F=0
100 IF EOF(1) THEN 140
110 INPUT#1,M(F)
120 F=F+1
130 GOTO 100
140 CLOSE#1:CLS
150 PRINT"collegare una EPROM vergine
160 PRINT"di almeno (. . .) byte
170 PRINT"indi premere ENTER
180 INPUT Z$:
190 PRINT"applic la tensione di +5V, premere ENTER
200 INPUT Z$:
210 PRINT"applicare Vpp, quindi premere ENTER
220 PRINT:PRINT" ATTENZIONE AL VALORE DI Vpp!
230 INPUT Z$:
240 FOR G=0 TO F-1
250 PRINT G,:D=M(G):PRINT D
260 OUT 888,D
270 OUT 890,1
280 FOR T=1 TO 39:NEXT T
290 OUT 890,0
300 NEXT G
310 PRINT:PRINT:PRINT:BEEP
320 PRINT"INTERROMPERE Vpp e premere ENTER
330 INPUT Z$:PRINT
340 PRINT"INTERROMPERE I +5V e premere ENTER
350 INPUT Z$:
360 CLS:PRINT"RIPRENDERE L'EPROM
    
```

Figura 6.

```

10 REM routine di programmazione
20 REM D = byte da programmare
30 OUT 888,D
40 REM trasferisce il byte sulla linea dei dati
della stampante
50 REM (porta di uscita dati stampante in 888)
60 OUT 890,1
70 REM inizio impulso di programmazione
80 REM (STROBE negato = bit D0 della porta 890)
90 FOR T=1 TO 39:NEXT T
100 REM temporizzazione 50 ms
110 REM (valore per COMMODORE PC1)
120 OUT 890,0
130 REM fine impulso di programmazione
140 REM e avanz.del contatore degli indirizzi
    
```

Figura 7.

```

1 REM ---- SAIROM ----
10 CLS
20 PRINT"nome del file da creare?
30 INPUT F$:F$=F$+" ROM"
40 OPEN"0",#1,F$
50 CLS:ON ERROR GOTO 100
60 PRINT" file";F$;" in corso di creazione"
70 READ D
80 PRINT#1,D;
90 GOTO 70
100 BEEP:CLS
110 PRINT F$;" è registrato su disco
120 END
130 REM (c)1989 Patrick GUEULLE
140 REM ---- seguono le linee DATA ----
    
```

Figura 8.

tarlo ad altre macchine:

- indirizzo della porta di uscita dei dati (nel nostro caso 888)
- indirizzo della porta di uscita dello STROBE (nel nostro caso 890)
- temporizzazione degli impulsi di programmazione (nel nostro caso 39 unità in 50 ms).

Naturalmente, è indispensabile un controllo con l'oscilloscopio prima di installare qualsiasi memoria (il programmatore può funzionare senza alcun problema con lo zoccolo vuoto). Attenzione soprattutto ad ottenere un impulso positivo di 50 ms su S ed uno negativo su S negato.

Il programma di Figura 6, scritto per un PC compatibile, va a cercare i dati da programmare su un file numerico presente sul dischetto. Durante la lettura riempie un array DIM M(4096), nel quale potrà rileggere i dati in fase di pro-

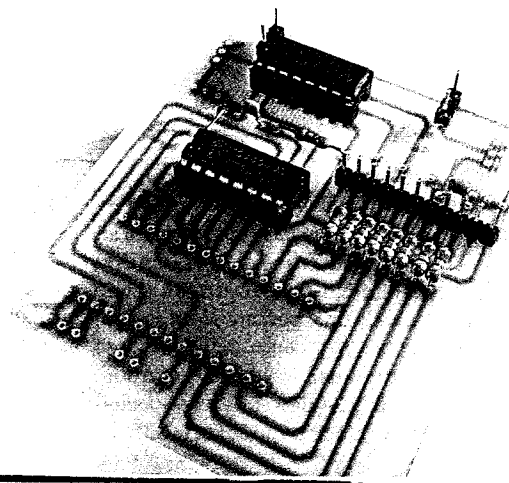
grammazione, senza subire interruzioni in grado di distruggere la EPROM per un prolungamento eccessivo dell'impulso di 50 ms.

grammazione, senza subire interruzioni in grado di distruggere la EPROM per un prolungamento eccessivo dell'impulso di 50 ms.

Il breve programma di Figura 8 permette di creare il file in questione, a partire da una serie di linee DATA contenenti i valori (decimali, esadecimali od ottali, espressi mediante gli appropriati prefissi) dei byte da programmare. Questo metodo di memorizzazione a partire da un listato permette di apportare facilmente qualsiasi modifica, senza dover reimpostare tutti i dati (questo vantaggio risulta particolarmente gradito quando si tratta di qualche centinaio o migliaia di dati!).

L'utilizzatore è ovviamente libero di riempire DIM M(4096) con dati di qual-

siasi provenienza (ad esempio una zona di memoria dove sarà stato precedentemente assemblato un programma in linguaggio macchina). Vedremo, in un prossimo articolo, come leggere questi dati in una EPROM standard. In ogni caso, prevedete di dover attendere poco meno di sei minuti per la programmazione di una 2716 (2 Kbyte) nel caso di un PC con clock di 4,77 MHz (ma ne esistono di più veloci!). Calcolate il doppio per una 2732 ma all'occorrenza, utilizzando un linguaggio diverso dal BASIC, potrete snellire notevolmente le operazioni.



molta precisione). All'occorrenza, si potranno facilmente costruire adattatori specifici per programmare i microprocessori "monochip" con EPROM incorporata. Avete ora a disposizione un attrezzo davvero universale, perché personalizzabile: può fare molto, ma richiede una certa attenzione. Cercate di non sbagliare nella "personalizzazione": consultate se necessario i data book del produttore (specialmente per le recenti EPROM CMOS 27C16 e seguenti) e potrete programmare qualsiasi EPROM, ad un costo insignificante.

© R.P. 501

28 Piedini	27256	27128	2764	2732	2716	24 Piedini
1	Vpp	Vpp	Vpp	-	-	-
2	A12	A12	A12	-	-	-
20	\bar{S}	GND	GND	\bar{S}	S	18
22	+5V	+5V	+5V	Vpp	+5V	20
23	A11	A11	A11	A11	Vpp	21
26	A13	A13	-	+5V	+5V	24
27	A14	\bar{S}	\bar{S}	-	-	-
28	+5V	+5V	+5V	-	-	-

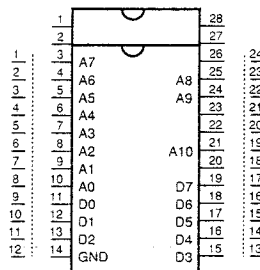


Figura 9.

Collegamenti alle normali EPROM

La tabella di Figura 9 indica i collegamenti da effettuare per programmare i cinque tipi di EPROM più comuni, effettuando, per i modelli più "capienti" alla 2732, le manovre descritte in Figura 2. La maggior parte delle altre EPROM sono compatibili con questo programmatore: basta rispettare le specifiche del produttore, soprattutto per quanto riguarda il valore preciso di Vpp (12-25 V, da rispettare con uno scarto di 0,5 V), la piedinatura e le caratteristiche dell'impulso di programmazione (polarità e durata, che può andare da 10 a 50 ms circa e deve essere rispettata con

ELENCO COMPONENTI

- R1 resistore da 8,2 k Ω 1/8 W 5%
 - R2/9 resistori da 470 Ω 1/8 W 5%, oppure rete DIL
 - C1-2 condensatori ceramici da 0,1 μ F
 - IC1 integrato CD4040
 - IC2 integrato CD4049
 - 2 interruttori a slitta
 - 1 zoccolo per IC a 28 piedini
- non compresi in kit:
- 1 connettore "Centronics" cavo a 11 conduttori
 - alimentatore 5 V
 - alimentatore 0-25 V
 - 18 spinotti a saldare, in barretta filo rigido da 6/10